Práctica #2: Uso de Variables y Arreglos

Introducción

El alumno creará programas en java que manejen variables y practicará la creación e implementación de arreglos. Para hacer esto se debe recordar lo visto con respecto a los tipos de identificadores que usa Java, como es la declaración de variables. Un ejemplo de cómo se declaran variables en java es el siguiente:

```
//Ejemplo de Uso de Variables
//(nombre alumno)
public class UsoVariables{
    public static void main(String args[]) {
       String nombre,apellido_paterno;
       nombre = "Manuel";
       apellido_paterno="Suarez";
       int edad = 28;
       System.out.println("Nombre: " + nombre + " " + apellido_paterno);
       System.out.println("Edad: " + edad);
    }
}
```

Un ejemplo de cómo se declara un arreglo en java es el siguiente:

```
//Ejemplo de Uso de Arreglos
//(nombre alumno)
public class UsoArray{
    public static void main(String args[]) {
       int arreglo[] = new int[5]; //se declara un arreglo vacio
       String nombres[] = {
       "Alfredo", "Hugo", "Laura", "Mariana", "Manuel"
       };
       System.out.println("Nombres: ");
       System.out.println(nombres[0]);
       System.out.println(nombres[1]);
       System.out.println(nombres[2]);
       System.out.println(nombres[3]);
       System.out.println(nombres[4]);
    }
}
```

Fundamento Teórico

Los identificadores nombran variables, funciones, clases, y objetos; cualquier cosa que queramos identificar o usar. Se caracterizan por comentar con una letra, guion bajo (_) o un símbolo de peso ($), de ahí los siguientes caracteres pueden ser letras o dígitos. Se distinguen mayúsculas y minúsculas y no hay longitud máxima.

```
tipo identificador [= valor], [identificador [= valor]];
int variable;
char nombre, apellido_paterno;
float $cantidad;
```

En Java se utilizan cinco tipos de elementos:

- Enteros: byte (8bits), short (16 bits), int 32(bits), long (64 bits)
- Reales con punto flotante: float (32 bits), double (64 bits)
- Booleanos: booleand (True, False)
- Caracteres: char
- Cadenas: String

Ahora para la creación de arreglos, como en el caso de una variable hay que declararlo antes de utilizar. Para declarar un arreglo se debe escribir la siguiente sintaxis:

```
TipoElemento[ ]  = nombreArreglo = new TipoElemento [tamañoArreglo]
char s[];
int Arreglo[] = new int[50];
int Tabla[][] = new int[4][5];
```

Crear un arreglo ya con variables

```
String nombres [] = {“Juan”,”Pedro”,”Hugo”,”Luis”};
```

Lo que es equivalente a:

```
String nombres[];
nombres = new String[4];
nombres[0] = new String (“Juan”);
nombres[1] = new String (“Pedro”);
nombres[2] = new String (“Hugo”);
nombres[4] = new String (“Luis”);
```

No se pueden crear arreglos estáticos en tiempo de compilación:

```
int lista[50]; //Generará un error en tiempo de compilación
```

Tampoco se puede rellenar un arreglo sin declarar el tamaño con el operador "new":

```
int lista[];
for (int i=o; i<9; i++)
      lista[i] = I;
```

Todos los arreglos en Java cuentan con el atributo "**length**", el cual se utiliza para conocer la longitud del arreglo.

```
int a[][] = new int[10][3];
a.length;   //10
a[0].length;  //3
```

Objetivos de Aprendizaje

El propósito de esta práctica es que el estudiante se familiarice con el entorno de la programación en Java, mediante el uso de comandos básicos al utilizar los diferentes tipos de identificadores para declarar variables, funciones, clases, y objetos; cualquier cosa que queramos identificar o usar.

Descripción de la práctica

1. Crear dentro de la carpeta "**Programación**" otra carpeta con el nombre de "**Practica2**".
2. Realizar el código fuente de "**UsoVariables**", utilizando el block de notas.
3. Guardar el archivo como **.java** en la carpeta de "**Practica2**".
4. Compilar el archivo mediante la línea de comandos (en ejecutar escribimos CMD) posicionándonos en la carpeta donde guardamos nuestro código fuente y escribimos "**javac nombrearchivo.java**".
5. En caso que el compilador marque un error, corregir el código fuente, y compilar nuevamente.
6. Si no hay error en el código fuente, ejecutar el programa escribiendo en la línea de comandos "**java nombrearchivo**".
7. Abrir un nuevo documento en el block de notas y crear un nuevo programa en el cual se declaren variables de punto flotante (double o float) y se impriman en pantalla los datos.

8. Guardar el archivo como "**UsoFlotante**" **.java** en la carpeta de "**Practica2**".
9. Compilar el archivo mediante la línea de comandos, en caso que el compilador marque un error, corregir y compilar nuevamente, si no hay error, ejecutar el programa.
10. Abrir un nuevo documento en el block de notas y realizar el código fuente de "**UsoArray**"
11. Guardar el archivo como **.java** en la carpeta de "**Practica2**".
12. Compilar el archivo mediante la línea de comandos, en caso que el compilador marque un error, corregir y compilar nuevamente, si no hay error, ejecutar el programa.
13. Realizar un reporte con los resultados obtenidos. El reporte debe contener Número de práctica, nombre de la experiencia educativa, nombre del alumno, matrícula, fecha de elaboración, desarrollo de la práctica, y resultados.

Materiales a emplear en la práctica

- Computadora con el JDK instalado.
- Block de notas.
- Línea de comandos o terminal.

Bibliografía

Froute Quintas, A. (2008). JAVA 2: Manual de usuario y tutorial (5th ed.). México: Alfaomega, ISBN 978-970-15-1430-6.